

Interference Search

EAST Search History

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	28	(heap and lock).clm.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/05/30 10:20

EAST Search History

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
S1	89	garbage near collect\$5 same heaps	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/05/30 08:00
S2	8	((("5,680,582") or ("6,412,053") or ("6,453,403") or ("6,175,900"))). PN.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/03/19 18:04
S3	2	(compact\$5 coalesc\$5) same lock\$3 same heaps	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/05/24 13:03
S4	11	(compact\$5 coalesc\$5) same heaps and lock	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/05/24 13:03
S5	5	garbage near collect\$5 same heaps same lock	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/05/24 15:21
S6	8	((("5,680,582") or ("6,412,053") or ("6,453,403") or ("6,175,900"))). PN.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/05/30 08:01
S7	1	S6 and lock	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/05/30 08:01
S8	1	S6 and (lock freeze)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/05/30 08:01


[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

 Search: ☒ The ACM Digital Library ☐ The Guide

heap <near> lock


[Feedback](#) [Report a problem](#) [Satisfaction survey](#)

 Terms used **heap near lock**

Found 3,719 of 177,263

Sort results by

relevance


[Save results to a Binder](#)
[Try an Advanced Search](#)

Display results

expanded form


[Search Tips](#)
[Try this search in The ACM Guide](#)
☐ Open results in a new window

Results 1 - 20 of 200

 Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

Best 200 shown

 Relevance scale ☐ ☐ ☐ ☐

1 [A comparative study of parallel and sequential priority queue algorithms](#)



Robert Rönngren, Rassul Ayani

 April 1997 **ACM Transactions on Modeling and Computer Simulation (TOMACS)**, Volume 7 Issue 2

Publisher: ACM Press

Full text available: pdf(640.10 KB)

 Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

Priority queues are used in many applications including real-time systems, operating systems, and simulations. Their implementation may have a profound effect on the performance of such applications. In this article, we study the performance of well-known sequential priority queue implementations and the recently proposed parallel access priority queues. To accurately assess the performance of a priority queue, the performance measurement methodology must be appropriate. We use the Classic ...

Keywords: parallel access priority queue, pending event set implementations, priority queue

2 [Scalable concurrent priority queue algorithms](#)



Nir Shavit, Asaph Zemach

 May 1999 **Proceedings of the eighteenth annual ACM symposium on Principles of distributed computing**

Publisher: ACM Press

Full text available: pdf(1.35 MB)

 Additional Information: [full citation](#), [references](#), [index terms](#)

3 [Scalable lock-free dynamic memory allocation](#)



Maged M. Michael

 June 2004 **ACM SIGPLAN Notices, Proceedings of the ACM SIGPLAN 2004 conference on Programming language design and implementation PLDI '04**, Volume 39 Issue 6

Publisher: ACM Press

Full text available: pdf(213.94 KB)

 Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Dynamic memory allocators (malloc/free) rely on mutual exclusion locks for protecting the consistency of their shared data structures under multithreading. The use of locking has many disadvantages with respect to performance, availability, robustness, and


[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

 Search: ☒ The ACM Digital Library ☐ The Guide



THE ACM DIGITAL LIBRARY


[Feedback](#) [Report a problem](#) [Satisfaction survey](#)

 Terms used **garbage near collection near lock**

Found 33,356 of 177,263

Sort results by


[Save results to a Binder](#)

 Try an [Advanced Search](#)

Display results


[Search Tips](#)

 Try this search in [The ACM Guide](#)
☐ Open results in a new window

Results 1 - 20 of 200

 Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

Best 200 shown

 Relevance scale ☐ ☐ ☐ ☐ ☐

1 [Parallel generational garbage collection](#)



Ravi Sharma, Mary Lou Soffa

 November 1991 **ACM SIGPLAN Notices, Conference proceedings on Object-oriented programming systems, languages, and applications OOPSLA '91**, Volume 26 Issue 11

Publisher: ACM Press

 Full text available: [pdf \(1.98 MB\)](#)

 Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

2 [Analysis of an algorithm for real time garbage collection](#)



Philip L. Wadler

 September 1976 **Communications of the ACM**, Volume 19 Issue 9

Publisher: ACM Press

 Full text available: [pdf \(1.04 MB\)](#)

 Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#)

A real time garbage collection system avoids suspending the operations of a list processor for the long times that garbage collection normally requires by performing garbage collection on a second processor in parallel with list processing operations, or on a single processor time-shared with them. Algorithms for recovering discarded list structures in this manner are presented and analyzed to determine sufficient conditions under which the list processor never needs to wait on the collector ...

Keywords: Lisp, analysis of algorithms, data structures, garbage collection, list processing, multiprocessing, parallel processing, real time, storage reclamation, time-sharing

3 [Garbage collection for a client-server persistent object store](#)



Laurent Amsaleg, Michael J. Franklin, Olivier Gruber

 August 1999 **ACM Transactions on Computer Systems (TOCS)**, Volume 17 Issue 3

Publisher: ACM Press

 Full text available: [pdf \(267.18 KB\)](#)

 Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

We describe an efficient server-based algorithm for garbage collecting persistent object stores in a client-server environment. The algorithm is incremental and runs concurrently with client transactions. Unlike previous algorithms, it does not hold any transactional locks on data and does not require callbacks to clients. It is fault-tolerant, but performs

[Home](#) | [Login](#) | [Logout](#) | [Access Information](#) | [Alerts](#) |

Welcome United States Patent and Trademark Office

Search Results[BROWSE](#)[SEARCH](#)[IEEE XPLORE GUIDE](#)

Results for "((heap <and> lock)<in>metadata)"

Your search matched 1 of **1351285** documents.☒ e-mailA maximum of 100 results are displayed, 25 to a page, sorted by **Relevance** in **Descending** order.

» Search Options

[View Session History](#)[New Search](#)

Modify Search

☐ Check to search only within this results setDisplay Format: ☒ Citation ☐ Citation & Abstract

» Key

IEEE JNL IEEE Journal or Magazine

IEEE JNL IEE Journal or Magazine

IEEE CNF IEEE Conference Proceeding

IEEE CNF IEE Conference Proceeding

IEEE STD IEEE Standard

[Select All](#) [Deselect All](#)

- ☐ 1. **Priority queues and sorting methods for parallel simulation**
Grammatikakis, M.D.; Liesche, S.;
[Software Engineering, IEEE Transactions on](#)
Volume 26, Issue 5, May 2000 Page(s):401 - 422
Digital Object Identifier 10.1109/32.846298
[AbstractPlus](#) | [References](#) | Full Text: [PDF](#)(5408 KB) IEEE JNL
[Rights and Permissions](#)

Indexed by
 Inspec®[Help](#) [Contact Us](#) [Privacy & :](#)

© Copyright 2006 IEEE --


[Home](#) | [Login](#) | [Logout](#) | [Access Information](#) | [Alerts](#) |

Welcome United States Patent and Trademark Office

Search Results

[BROWSE](#)[SEARCH](#)[IEEE XPLORE GUIDE](#)

Results for "((garbage <near> collection <and> lock)<in>metadata)"

☒ e-mail

Your search matched 5 of 1351285 documents.

A maximum of 100 results are displayed, 25 to a page, sorted by Relevance in Descending order.

» Search Options

[View Session History](#)[New Search](#)

Modify Search

((garbage <near> collection <and> lock)<in>metadata)

☐ Check to search only within this results setDisplay Format: ☒ Citation ☐ Citation & Abstract

» Key

IEEE JNL IEEE Journal or Magazine

IEEE JNL IEE Journal or Magazine

IEEE CNF IEEE Conference Proceeding

IEEE CNF IEE Conference Proceeding

IEEE STD IEEE Standard

[Select All](#) [Deselect All](#)

- ☐ 1. **Lock-free garbage collection for multiprocessors**
Herlihy, M.P.; Moss, J.E.B.;
[Parallel and Distributed Systems, IEEE Transactions on](#)
Volume 3, Issue 3, May 1992 Page(s):304 - 311
Digital Object Identifier 10.1109/71.139204
[AbstractPlus](#) | Full Text: [PDF\(732 KB\)](#) IEEE JNL
[Rights and Permissions](#)
- ☐ 2. **Efficient and reliable lock-free memory reclamation based on reference c**
Gidenstam, A.; Papatriantafyllou, M.; Sundell, H.; Tsigas, P.;
[Parallel Architectures, Algorithms and Networks, 2005. ISPAN 2005. Proceedin](#)
[International Symposium on](#)
7-9 Dec. 2005 Page(s):6 pp.
Digital Object Identifier 10.1109/ISPAN.2005.42
[AbstractPlus](#) | Full Text: [PDF\(200 KB\)](#) IEEE CNF
[Rights and Permissions](#)
- ☐ 3. **A fast analysis for thread-local garbage collection with dynamic class loa**
Jones, R.; King, A.C.;
[Source Code Analysis and Manipulation, 2005. Fifth IEEE International Works](#)
30 Sept.-1 Oct. 2005 Page(s):129 - 138
Digital Object Identifier 10.1109/SCAM.2005.1
[AbstractPlus](#) | Full Text: [PDF\(296 KB\)](#) IEEE CNF
[Rights and Permissions](#)
- ☐ 4. **Impact of JIT/JVM optimizations on JAVA application performance**
Shiv, K.; Iyer, R.; Newburn, C.; Dahlstedt, J.; Lagergren, M.; Lindholm, O.;
[Interaction Between Compilers and Computer Architectures, 2003. INTERACT](#)
[Proceedings, Seventh Workshop on](#)
8 Feb. 2003 Page(s):5 - 13
[AbstractPlus](#) | Full Text: [PDF\(543 KB\)](#) IEEE CNF
[Rights and Permissions](#)
- ☐ 5. **On mixing queries and transactions via multiversion locking**
Bober, P.M.; Carey, M.J.;
[Data Engineering, 1992. Proceedings, Eighth International Conference on](#)
2-3 Feb. 1992 Page(s):535 - 545
Digital Object Identifier 10.1109/ICDE.1992.213155